

CURSO: ENGENHARIA DE SOFTWARE**DISCIPLINA:** Estruturas de Dados e Algoritmos**TURMA****B****SEMESTRE:** 2014.2**CRÉDITOS:****4****PROFESSOR:** Fernando William Cruz

2a. PROVA INDIVIDUAL

1) (3,5 pontos) Um programa público de distribuição de alimentos funciona da seguinte maneira: a cada quinzena, um caminhão com cestas básicas chega ao centro do município. Uma fila é formada pelos cidadãos, que podem trocar um *ticket* por uma cesta. Embora não haja restrições quanto ao número de *tickets* que um mesmo cidadão pode ter, ele pode trocar apenas um *ticket* por vez: caso queira, ele pode retornar ao final da fila para trocar outras vezes. Por outro lado, o caminhão tem um número limitado de cestas, de modo que a distribuição se encerra com a entrega da última cesta.

Dada a quantidade de cestas que o caminhão disponibilizou e as informações sobre a fila (nome do cidadão e a quantidade de *tickets* que possui), determine o nome do cidadão que recebeu a última cesta. Considere que se um cidadão tinha mais de um *ticket*, ele retornou ao fim da fila imediatamente após receber sua cesta.

Entrada: A entrada consiste em uma linha com o número de cestas disponíveis, o número de cidadãos N na fila e as informações de cada cidadão: o primeiro nome e quantidade de *tickets*. Todas as informações são inteiros positivos, exceto o primeiro nome do cidadão, que é composto apenas de caracteres alfabéticos.

Saída: A saída do programa deverá ser a mensagem “C RECEBEU A ULTIMA CESTA”, onde C é o primeiro nome do cidadão que recebeu a última cesta, ou “AS CESTAS ATENDERAM A TODOS OS CIDADAOs”, caso a fila se encerre antes de acabar as cestas disponíveis. Ao final da mensagem deve ser impressa uma quebra de linha.

Entradas	Saídas
1 2 JOAO 2 MARIA 1	JOAO RECEBEU A ULTIMA CESTA
3 2 JOAO 2 MARIA 1	JOAO RECEBEU A ULTIMA CESTA
10 4 JOAO 2 MARIA 1 PEDRO 3 LUCAS 1	AS CESTAS ATENDERAM A TODOS OS CIDADAOs
6 4 JOAO 2 MARIA 1 PEDRO 3 LUCAS 1	PEDRO RECEBEU A ULTIMA CESTA

2) (3,5 pontos) O chefe do almoxarifado de uma pequena empresa deu a seguinte instrução aos seus funcionários: uma caixa só pode ser colocada em cima de uma caixa que contém massa maior ou igual a caixa a ser empilhada. As caixas a serem empilhadas têm mesmas dimensões e são fabricadas com mesmo material, de modo que são diferenciadas apenas pelas etiquetas que indicam a massa do material armazenado na caixa. Com a chegada de uma nova remessa de material, os funcionários reservaram o espaço necessário para uma nova pilha de caixas.

Um funcionário trazia as caixas do caminhão para a porta do almoxarifado, enquanto um segundo funcionário consultava a etiqueta da caixa recém-chegada: se a massa contida nela permitisse o empilhamento, o funcionário a colocava na pilha; caso contrário, separava a caixa numa sala ao lado. Sabendo que a pilha pode ter, no máximo, seis caixas, e conhecida a ordem em que as caixas foram sendo retiradas do caminhão e analisadas, determine a massa total que o funcionário conseguiu empilhar.

Entrada: A entrada consiste em uma linha com a quantidade N de caixas que serão retiradas do caminhão e as massas, em ordem de retirada, destas N caixas. Pode-se considerar todos os valores são inteiros positivos e que a massa das caixas seja dada em kilogramas.

Saída: A saída do programa deverá ser a mensagem “MASSA EMPILHADA: M kg”, onde M é a massa total de material contido nas caixas que foram empilhadas, seguida de uma quebra de linha.

Entradas	Saídas
1 12	MASSA EMPILHADA: 12 kg
3 10 10 15	MASSA EMPILHADA: 20 kg
6 30 20 25 12 20 8	MASSA EMPILHADA: 70 kg
10 40 25 30 25 20 8 10 2 2 1	MASSA EMPILHADA: 120 kg

3) (3,0 pontos) Monte uma calculadora que receba expressões na notação infixa e mostre o resultado, conforme apresentado na tabela abaixo:

Entradas	Saídas
$(2+3*4)$	14
$(2*(3+4)/7-6)$	-4
$(7+(6-(5+(4-(3+2))))))$	9
$(8+7*6/3*4-5)$	59

Para facilitar a construção do programa, considere que:

- operandos são formados por dígitos entre 0 e 9
- operadores possíveis são +, -, * e /
- toda expressão de entrada está envolta entre parêntesis e está correta
- espaços em branco entre operandos e operadores devem ser descartados.

Obs.: A pilha deve ser implementada utilizando lista encadeada.

Boa prova!